Compilers

Final Project

Hand in the printed source code, test cases, and sample compiled output at the beginning of the Final Exam meeting, during which you will make a short presentation.

Write a full compiler (lex, parse, type-checking, and code generation) for a grammar as expressive as that of project 2, with the following required additions:

- while loops
- comments
- if-then

The generated code must conform to the 6502a instructions set specified on our class web site and execute on one of the 6502a-based operating systems also found on our class web site.

If you're feeling up to it, consider adding one or more of the following for extra credit and coolness:

- sub program call and return (procedure)
- sub program call and return (function)
- integer arrays
- multiplication and division

As with the all of the projects before this,

- Include verbose output functionality that traces the stages of the parser including the construction of the symbol table and type checking actions.
- When you detect an error report it in helpful detail including where it was found.

Create several test programs that cause as many different types of errors that you can think of in order to thoroughly test your code. Include a several test cases that show it working as well. Write up your testing results in a document and hand them in.

alan@Labosueur.com
www.Labouseur.com